Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	1570	interrupt$4 with ((plurality multiple several) near3 (processor cpu))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/22 08:00
S2	20	S1 same (response with determin$5)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/16 12:33
S3	0	S2 same defer$4	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/15 13:48
S4	3	S2 and defer$4	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/15 13:48
S5	46	nvidia.as. and interrupt and (multiple plurality several)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/16 12:44
S6	846	interrupt$4 WITH ((multiple OR plurality OR several) adj2 (processor OR cpu))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/17 16:10
S7	11	S6 SAME (response WITH determin$5)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/17 16:27
S8	0	"deferred servicing procedure"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT	OR	OFF	2005/11/17 16:35
S9	68	"deferred procedure call"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT	OR	OFF	2005/11/18 11:39
S10	0	"deferred procedure call" and ((multiple OR plurality OR several) adj2 (processor OR cpu))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT	OR	OFF	2005/11/18 11:40
S11	7	"deferred procedure call" and ((multiple OR plurality OR several) near3 (processor OR cpu))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT	OR	OFF	2005/11/18 11:40

S12	1571	interrupt$4 with ((plurality multiple several) near3 (processor cpu))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/22 08:00
S13	3	S12 same deferred	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/22 08:00
S14	0	"private register mapping"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/22 10:42
S15	477	"register mapping"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/22 10:42
S16	22	S15 with interrupt	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/22 10:43
S17	2	S16 with updat$4	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/22 10:42
S18	15	(US-20010029556-$ or US-20020144004-$ or US-20020161957-$ or US-20030101293-$ or US-20030135787-$ or US-20030140179-$ or US-20030187914-$ or US-20040111549-$ or US-20040122986-$ or US-20040122997-$).did. or (US-5446910-$ or US-6282601-$ or US-6571206-$ or US-6813665-$ or US-6772189-$).did.	US-PGPUB; USPAT	OR	OFF	2005/11/23 10:26
S19	0	S18 and (register near1 (map$4 table))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/22 11:38
S20	2	S18 and (register near2 (map$4 table))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/11/23 08:14
S21	1321	interrupt same (register near2 (map$4 table))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/23 08:15

S22	37	interrupt same (register adj mapping)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/23 09:24
S23	2	"5867687".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/11/23 09:24
S24	15	(US-20010029556-$ or US-20020144004-$ or US-20020161957-$ or US-20030101293-$ or US-20030135787-$ or US-20030140179-$ or US-20030187914-$ or US-20040111549-$ or US-20040122986-$ or US-20040122997-$).did. or (US-5446910-$ or US-6282601-$ or US-6571206-$ or US-6813665-$ or US-6772189-$).did.	US-PGPUB; USPAT	OR	OFF	2005/11/23 10:26
S25	4	S24 and graphics	US-PGPUB; USPAT	OR	OFF	2005/11/23 10:27
S26	1	"6691180".pn.	US-PGPUB; USPAT	OR	OFF	2005/11/23 10:28

WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Wednesday, November 23, 2005

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR	
☐	L6	L5.ti.	17
☐	L5	(interrupt$4 near5 (handl$4 or servic$4 or enabl$4 or disabl$4)) with (priorit$7 near5 level)	588
☐	L4	L3.ti.	121
☐	L3	(interrupt$4 with priorit$7 with level)	2702
☐	L2	L1 same interrupt$4	0
☐	L1	(private near2 map$4) same ((read$4 or access$4) near3 (register or memory or buffer or queue or file))	34

END OF SEARCH HISTORY

L1: Entry 17 of 34 File: USPT Sep 13, 2005

DOCUMENT-IDENTIFIER: US 6944662 B2
TITLE: System and methods providing automatic distributed data retrieval, analysis and reporting services

Detailed Description Text (329):
Another important aspect of the system 1 file sharing is that the storage may be on a mapped drive. Those skilled in the art will appreciate that it is not a simple task to provide shared access to files on such a mapped drive using conventional "private folders", which are accessible only to the owner who connects to the mapped drive using a password. The subject system 1 allows publishers to specify files in private, mapped folders and to give subscribers access to these files by relaying their requests to obtain them to the file server, as though the owner were himself making the request. This procedure is carried out invisibly to the subscriber and publisher.

First Hit Previous Doc Next Doc Go to Doc#

Generate Collection Print

L1: Entry 1 of 34 File: PGPB Nov 3, 2005

DOCUMENT-IDENTIFIER: US 20050246705 A1
TITLE: Method for dynamically allocating and managing resources in a computerized system having multiple consumers

Detail Description Paragraph:
[0060] Mapping--private or shared in the case of RW. If it is shared, only the creator of the shared storage should be charged for this memory. The term private refers to a memory that only a specific program can access, such as memory that was allocated when the specific program started running, or that was "malloc"ed. The term Shared refers to one that is shared between processes, for example, while loading a shared object (e.g., a Dynamic Link Library (DLL, which is a collection of small programs, each of which can be called when needed by a larger program that is running in the computer, in Windows2000 environment.

Previous Doc Next Doc Go to Doc#

First Hit Previous Doc Next Doc Go to Doc#

Generate Collection Print

L1: Entry 2 of 34 File: PGPB Nov 3, 2005

DOCUMENT-IDENTIFIER: US 20050246502 A1
TITLE: Dynamic memory mapping

Abstract Paragraph:
In at least some embodiments, a system comprises two processor cores, an external memory coupled to the two processor cores, and a program that is executable at least in part by one or both of the processing cores. When executed by one of the processor cores the program causes the processor core to map a private region of the external memory, which is accessible only to one of the two processor cores, to a pre-reserved region of memory addresses used by the other processor core. The mapping permits the processor core that does not have direct access to the private memory region of the other processor core to access data stored in the private region. In at least some embodiments, the mapped memory can be subsequently unmapped and re-mapped to another private memory region at run-time.

Summary of Invention Paragraph:
[0007] In at least some embodiments, a system comprises two processor cores, an external memory coupled to the two processor cores, and a program that is executable at least in part by one or both processing cores. When executed by one of the processor cores, the program causes the processor core to map a private region of the external memory, which is accessible only to one of the two processor cores, to a pre-reserved region of memory addresses used by the other processor core. The mapping permits the processor core that does not have direct access to the private memory region of the other processor core to access data stored in the private region.

CLAIMS:

3. The system of claim 2, wherein the program causes the entire private region of the external memory accessible by the first processor core to be mapped to the smaller pre-reserved region of memory addresses by mapping segments of the private region equal in size to the pre-reserved region, unmapping the mapped segment of the private region from the pre-reserved region, and mapping the next segment of the private region to the pre-reserved region, until the entire private region has been mapped.

5. The system of claim 4, wherein said program causes the entire private region of the external memory accessible by the first processor core to be mapped to the pre-reserved region of memory addresses by mapping a segment of the private region to a segment of the pre-reserved region equal in size to the segment of the private region, unmapping the mapped segment of the private region from the pre-reserved, and mapping the next segment of the private region to the next segment of the pre-reserved region.

Previous Doc Next Doc Go to Doc#

L1: Entry 4 of 34 File: PGPB Nov 4, 2004

DOCUMENT-IDENTIFIER: US 20040221290 A1
TITLE: Management of virtual machines to utilize shared resources

Detail Description Paragraph:
[0036] FIGS. 5(a) and 5(b) illustrate the foregoing operation of resource manager 217 in more detail. The resource manager 217 performs the steps of FIGS. 5(a) and 5 (b) for each user virtual machine 12, 14 and 16 to determine the need to create a clone of the virtual machine or delete a clone of the virtual machine. Initially, the resource manager 217 sets new soft limits for the real CPU, the virtual private memory and the real I/O allocated to the virtual machine or reads the original ones from the directory 191 if available there (step 100). The real CPU allocation for a virtual machine is the amount of CPU processing time available to the virtual machine. The virtual private memory is the amount of private memory allocated to the virtual machine; the addresses assigned for this private memory are mapped to the real memory. The real I/O is the amount of real I/O bandwidth available to the virtual machine. Next, the resource manager fetches from the usage records 85 the current utilization levels of the foregoing resources by the virtual machines (step 102). As noted above, the common base portion periodically monitors these virtual and real utilization levels for the resources. Then, the resource manager reads the real CPU utilization level of the virtual machine (step 104). Next, the resource manager determines if the current real CPU utilization by the virtual machine is more than its soft limit (decision 106). If not, then the resource manager determines if the virtual machine needs additional virtual resources (decision 107). This determination is made by monitoring the workload of the virtual machine each time it has a time slice of the real CPU. If the virtual machine usually or always has outstanding work to complete when it has access to the real CPU, then it probably needs a greater share of the real CPU. If so, then the resource manager creates a clone of the virtual machine in the manner described above (step 108). Also, the resource manager gives the clone access to the shared work queue in the manner described above (step 110).

Hit List

First Hit | Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 10 of 17 returned.

☐ 1. Document ID: US 5867687 A

Using default format because multiple data bases are involved.

L6: Entry 1 of 17 File: USPT Feb 2, 1999

US-PAT-NO: 5867687
DOCUMENT-IDENTIFIER: US 5867687 A

TITLE: Microprocessor system for handling multiple priority levels interrupt requests to processor and interrupt process identifiers

DATE-ISSUED: February 2, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Simpson; Robert John	Bristol			GB

US-CL-CURRENT: 710/264

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 2. Document ID: GB 2381891 A

L6: Entry 2 of 17 File: EPAB May 14, 2003

PUB-NO: GB002381891A
DOCUMENT-IDENTIFIER: GB 2381891 A
TITLE: Testing interrupts having a service priority order by applying combinations of interrupt priority levels

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

☐ 3. Document ID: GB 2381890 A

L6: Entry 3 of 17 File: EPAB May 14, 2003

PUB-NO: GB002381890A
DOCUMENT-IDENTIFIER: GB 2381890 A
TITLE: Testing interrupts having a service priority order by applying different interrupt priority levels

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw De

4. Document ID: US 20050177668 A1

L6: Entry 4 of 17 File: DWPI Aug 11, 2005

DERWENT-ACC-NO: 2005-570645
DERWENT-WEEK: 200558

TITLE: Data processor in multiprocessor system, compares contiguous portions of priority level values of pending and active interrupt handling programs and accordingly controls order of activation of several pending interrupt handling programs

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

5. Document ID: JP 2004287838 A

L6: Entry 5 of 17 File: DWPI Oct 14, 2004

DERWENT-ACC-NO: 2004-714703
DERWENT-WEEK: 200470

TITLE: Interruption controller for computer, selects level handler, based on interruption factor bit or present interrupt level and priority of next interrupt level, and executes specified interruption program in interruption handler

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

6. Document ID: GB 2381891 B, GB 2381891 A, US 20030120975 A1

L6: Entry 6 of 17 File: DWPI Oct 29, 2003

DERWENT-ACC-NO: 2003-423699
DERWENT-WEEK: 200373

TITLE: Testing priority levels of microprocessor interrupt sources, by sorting sources, determining array of priority levels to be assigned, incrementing global counter, assigning array, enabling interrupts and transferring count value

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

7. Document ID: AU 2001284738 A8, WO 200241153 A2, AU 200184738 A, EP 1336139 A2, KR 2003051834 A, US 6681281 B1, CN 1474971 A, JP 2004521410 W

L6: Entry 7 of 17 File: DWPI Sep 8, 2005

DERWENT-ACC-NO: 2002-508343
DERWENT-WEEK: 200568

TITLE: Multi-level interrupt scheme, for a computer system, that allows a bus device to indicate an interrupt priority level to enable a bus controller to provide suitable arbitration

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

---

☐ 8. Document ID: JP 2002055845 A

L6: Entry 8 of 17 File: DWPI Feb 20, 2002

DERWENT-ACC-NO: 2002-233428
DERWENT-WEEK: 200229

TITLE: Parallel process type general purpose debug for multitasking system, collects contents of memory and register required for debug, based on starting of interruption handler with top-priority working level

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

---

☐ 9. Document ID: KR 453468 B, JP 11224201 A, DE 19904333 A1, CN 1228561 A, KR 99072432 A, JP 3097648 B2, TW 425507 A, US 6269419 B1

L6: Entry 9 of 17 File: DWPI Oct 20, 2004

DERWENT-ACC-NO: 1999-513798
DERWENT-WEEK: 200514

TITLE: Interruption control apparatus for information processor - determines priority level of interruption factor, generates interruption vector based on interruption factor and outputs interruption vector to enable pipeline processing in processor

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

---

☐ 10. Document ID: JP 09259066 A

L6: Entry 10 of 17 File: DWPI Oct 3, 1997

DERWENT-ACC-NO: 1997-541387
DERWENT-WEEK: 199750

TITLE: Input output controller for service code interruption - has service code interruption register and local memory that store service code and detailed service code data, respectively, after release from service code buffer in predetermined priority level

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw D

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L5.ti.	17

Display Format: - Change Format

Previous Page Next Page Go to Doc#